

REMARKS

Claims 1 and 3-20 were pending at the time of examination. Claims 1, 3-4, 8, 10, 14-15, and 17-19 have been amended. No claims have been canceled. No new matter has been added. The applicants respectfully request reconsideration based on the foregoing amendments and these remarks.

Claim Rejections – 35 U.S.C. § 101

Claims 1 and 3-9 were rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter. The last step of claim 1 has been amended to recite:

“when the first instance has not been identified in the one or more libraries, creating the first instance, and when the first instance has been identified in the one or more libraries, using the linker symbol name of the first instance as a reference to the first instance already contained within the one or more libraries, thereby avoiding duplication of instances and reducing the time needed for compiling the source program.”

That is, the claim recites a concrete, tangible, and useful result, not only for when the first instance has not been identified, but also for when the instance has been identified. The applicants submit that claim 1 and 3-9 comply with the “practical application test” as described by the Examiner in the Office Action, and that the rejection under 35 U.S.C. § 101 for claims 1 and 3-9 be removed.

Claim Rejections – 35 U.S.C. § 103

Claims 1, 3-7 and 9-20 were rejected under 35 U.S.C § 103(a) as being unpatentable over U.S. Patent No. 6,308,320 to Burch (hereinafter “Burch”) in view of U.S. Patent No. 5,408,665 to Fitzgerald (hereinafter “Fitzgerald”). Claim 8 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Burch and Fitzgerald, as applied to claim 1, and in further view of U.S. Patent No. 6,041,180 to Perks et al. (hereinafter “Perks”). The applicants respectfully traverse these rejections.

Claim 1 has been amended to more clearly define the invention and the distinctions over the Burch and Fitzgerald combination. As background, before addressing the specific limitations of claim 1, compilers generally accept a source file containing multiple data structure definitions and associated functions that can operate on the data structures, and produce an object file containing a realization, or instance, for each of those data structures and its functions. The form of those realizations or instances depends on the instructions presented to the compiler. In the

course of development, the same source file may be compiled with different instructions into different object files. The Burch invention discloses a mechanism for avoiding producing an object file (embodying both a source and compiler instructions) when such an object file has been produced before.

In modern programming languages, the same template and a set of operations that can be performed on the template (e.g., a “stack” data structure with the operations “push,” “pop,” “list,” and so on) may well appear in multiple source files, which when compiled may result in an instance of the template, i.e., a specialization of the template (such as a stack of integers) in each of multiple object files. The applicants’ invention, as defined in claim 1, is a mechanism to avoid producing multiple instances across object files and libraries. In essence, while Burch reduces the number of compilations by checking if an object file already exists in the repository, the applicants’ invention operates to reduce the amount of work in each compilation by checking if an instance exists in one or more library object files.

This issue can be further clarified in two respects. First, the applicants’ invention avoids the creation of multiple instances independently of the instructions given to the compiler. That is, the applicants’ invention, as defined in claim 1, avoids any duplicate instances, not just duplicate identical realizations, as is the case with Burch. Hence the use of linker symbol names in the applicants’ invention, in contrast to a hash of object file names and compiler instructions, in Burch. Second, the applicants’ invention, as defined in claim 1, is independent of the object structure. While it is generally true, as the Examiner states, that one can extract objects from libraries, it is false in the case of dynamic (shared) libraries. Furthermore, it is also irrelevant to the invention as defined in claim 1, because claim 1 does not describe any extraction of objects. On the contrary, the invention as defined in claim 1, concerns the recognition of an instance, not the recognition of an object. Once the instance is identified in a library, no more work need be done. The instance will appear in the compiled object file as a consequence of using the one or more libraries during compilation.

The applicants have amended claim 1 in order to clarify these differences with respect to Burch. For example, in order to avoid confusion with respect to the term “library of object files,” claim 1 has been amended to specify that the libraries are “libraries of instances.” The term “instance” has been defined in claim 1 as “a specialization of a template and its available operations” in accordance with what is set forth in the specification on page 2, lines 13-20. The last step of claim 1 has been amended to set forth that the first instance is created “when the first instance has not been identified in the one or more libraries,” that “the identified linker symbol name of the first instance” is used as “a reference to the instance already contained within the

library” when the first instance has been identified in the one or more libraries, and that this results in “avoiding duplication of instances and reducing the time needed for compiling the source program.” It is respectfully submitted that neither Burch nor Fitzgerald, alone or in combination, shows this combination of features, and for at least these reasons, the rejection of claim 1 is unsupported by the art and should be withdrawn.

Claims 3-9 are all dependent from claim 1, and the rejection of these claims is unsupported by the cited art for at least the reasons discussed above with regards to claim 1, and should be withdrawn.

In rejecting claim 8, the Examiner combines Burch and Fitzgerald with Perks to establish a *prima facie* case of obviousness. Perks was used by the Examiner since “Perks via a linking process also uses comparing of object files to a [sic] encoded scheme to identify whether some discrepancies exist before assembling the final executable, and further disclose templates” (Office Action, page 9, item 6). As stated by the Examiner, Perks compares object files to the encoded scheme. However, as was stated above, the invention as defined in claims 1 and 8 is not concerned with object files, but with instances. Thus, Perks does not add anything that cures the deficiencies discussed above, and it is respectfully submitted that the rejection of claim 8 in particular be withdrawn.

The Examiner rejected claim 14 using essentially the same rationale that was used in rejecting claim 1. Thus for at least these reasons presented above with respect to claim 1, the rejection of claim 14 is unsupported by the art and should be withdrawn.

Claims 15-16 both depend from claim 14, and the rejection of these claims is unsupported by the cited art for at least the reasons discussed above with regards to claim 14, and should be withdrawn.


Claim 17 is a *Beauregard* claim corresponding to claim 1. For reasons substantially similar to those set forth above with regards to claim 1, the applicant respectfully contends that the rejection of claim 17 is unsupported by the cited art and should be withdrawn.

Claims 18-20 all depend from claim 17, and are *Beauregard* claims corresponding to claims 3, 4, and 6, respectively. The rejection of these claims is unsupported by the cited art for at least the reasons discussed above with regards to claim 17, and should be withdrawn.

Conclusion

The applicants believe that all pending claims are allowable and respectfully request a Notice of Allowance for this application from the Examiner. Should the Examiner believe that a telephone conference would expedite the prosecution of this application, the undersigned can be reached at the telephone number set out below.

Respectfully submitted,
BEYER WEAVER & THOMAS, LLP


Fredrik Mollborn
Reg. No. 48,587

P.O. Box 70250
Oakland, CA 94612-0250
(650) 961-8300